Finding disjoint paths in networks with star shared risk link groups

Jean-Claude Bermond, David Coudert, Gianlorenzo d'Angelo, Fatima Zahra Moataz

Finding Disjoint Paths in Networks with Star Shared Risk Link Groups*

Jean-Claude Bermond[1,2], David Coudert[2,1], Gianlorenzo D'Angelo[3], and Fatima Zahra Moataz[1,2]

[1]Univ. Nice Sophia Antipolis, CNRS, I3S, UMR 7271, 06900 Sophia Antipolis, France
[2]Inria, France
[3]Gran Sasso Science Institute (GSSI), L'Aquila, Italy

Abstract

The notion of Shared Risk Link Groups (SRLG) has been introduced to capture survivability issues where some links of a network fail simultaneously. In this context, the *k-diverse routing* problem is to find a set of k pairwise SRLG-disjoint paths between a given pair of end nodes of the network. This problem has been proven NP-complete in general and some polynomial instances have been characterized.

In this paper, we investigate the k-diverse routing problem in networks where the SRLGs are localized and satisfy the *star property*. This property states that a link may be subject to several SRLGs, but all links subject to a given SRLG are incident to a common node. We first provide counterexamples to the polynomial time algorithm proposed by X. Luo and B. Wang (DRCN'05) for computing a pair of SRLG-disjoint paths in networks with SRLGs satisfying the star property, and then prove that this problem is in fact NP-complete. We then characterize instances that can be solved in polynomial time or are fixed parameter tractable, in particular when the number of SRLGs is constant, the maximum degree of the vertices is at most 4, and when the network is a directed acyclic graph.

Finally we consider the problem of finding the maximum number of SRLG-disjoint paths in networks with SRLGs satisfying the star property. We prove that this problem is NP-hard to approximate within $O(|V|^{1-\varepsilon})$ for any $0 < \varepsilon < 1$, where V is the set of nodes in the network. Then, we provide exact and approximation algorithms for relevant subcases.

Keywords: Diverse routing; Shared Risk Link Group; Colored graph; Complexity; Algorithms; Disjoint paths.

1 Introduction

To ensure reliable communications in connection oriented networks such as optical backbone networks, many protection schemes have been proposed. One of the most used, called *dedicated path protection*, consists in computing for each demand both a working and a protection path. A general requirement is that these paths have to be disjoint, so that at least one of them can survive a single failure in the network. This method works well in a single link failure scenario,

*This work is partially supported by a grant from the "Conseil régional Provence Alpes-Côte d'Azur", ANR project Stint under reference ANR-13-BS02-0007, ANR program "Investments for the Future" under reference ANR-11-LABX-0031-01, and by European project FP7 EULER (Grant No.258307), and the Research Grant 2010N5K7EB 'PRIN 2010' ARS TechnoMedia (Algoritmica per le Reti Sociali Tecno-mediate) from the Italian Ministry of University and Research.

as it consists in finding two edge-disjoint paths between a pair of nodes. This is a well-known problem in graph theory for which there exist efficient polynomial time algorithms [27, 28].

However, the problem of finding two disjoint paths between a pair of nodes becomes much more difficult, in terms of computational complexity, in case of multiple correlated link failures that can be captured by the notion of *Shared Risk Link Group* (or *SRLG*, for short). In fact, a SRLG is a set of network links that fail simultaneously when a given event (risk) occurs. The scope of this concept is very broad. It can correspond, for instance, to a set of fiber links of an optical backbone network that are physically buried at the same location and therefore could be cut simultaneously (i.e. backhoe or JCB fade). It can also represent links that are located in the same seismic area, or radio links in access and backhaul networks subject to localized environmental conditions affecting signal transmission, or traffic jam propagation in road networks. Note that a link can be affected by more than one risk. In practice, the failures are often localized and common SRLGs satisfy the *star property* [24] (coincident SRLGs in [10]). Under this property, all links of a given SRLG share an endpoint. Such failure scenarios can correspond to risks arising in router nodes like card failures or to the cut of a conduit containing links issued from a node (see Figure 1).

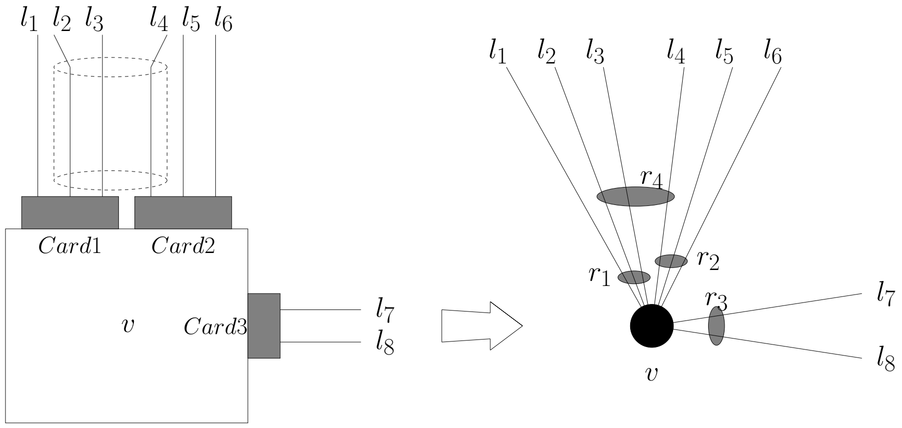


Figure 1: Example of localized risks: link l_4 shares risk r_2, corresponding to Card 2 failure, with links l_5 and l_6, and shares risk r_4, corresponding to a conduit cut, with links l_2 and l_3.

The graph theoretic framework for studying optimization problems in networks with SRLGs is the *colored graph* model [9, 32, 13, 10, 24, 7]. In this model, the network topology is modeled by a graph $G = (V, E)$ and the set of SRLGs by a set of colors $\mathcal{C}$. Each SRLG is modeled by a distinct color, and that color is assigned to all the edges corresponding to the network links subject to this SRLG. Also, an edge modeling a network link subject to several SRLGs will be assigned as many colors as SRLGs. A colored graph is therefore defined by the triple $(V, E, \mathscr{C})$, where $\mathscr{C}$ is a *coloring* function, $\mathscr{C} : E \to 2^{\mathcal{C}}$, that assigns a subset of colors to each edge. The colored graph model is also known as the *labeled graph* model [14]. Furthermore, some studies assumed that an edge is assigned at most one color [7, 14, 22]. Notice that the computational complexity of some optimization problems may be different in the model in which an edge is assigned at most one color than in the model in which it can be assigned multiple colors, and the impact of the transformation from one model to the other on problems complexities has been investigated in [8]. In the setting of colored graphs, the star property means that all the edges with a given color share a common vertex.

1.1 Related work

In the context of colored graphs, basic graph connectivity problems have been re-stated in terms of colors and proven much more difficult to address than their basic counterparts. For instance, the minimum color st-path problem is to find a path from vertex s to vertex t in the graph that minimizes the cardinality of the union of the colors of the edges along that path. This problem has been proven NP-hard [26, 4, 5] and hard to approximate [7, 18] in general, $W[2]$-hard when parameterized by the number of used colors and $W[1]$-hard when parameterized by the number of edges of the path [14]. However, it has been proven in [8] that the minimum color st-path problem can be solved in polynomial time in colored graphs with the star property. Other optimization problems on graphs have been studied in the context of colored graphs such as the minimum color cut [13, 7], the minimum color st-cut [7], the minimum color maximum matching [14] .

The *k-diverse routing* problem in presence of SRLGs consists in finding a set of k SRLG-disjoint paths between a pair of vertices (i.e. paths having no risk in common). Note that many authors use, in the case $k = 2$, *diverse routing* instead of 2-diverse routing. With no restriction on the graph structure and on the assignment of SRLGs to edges, even finding two SRLG-disjoint paths is NP-complete [21], and therefore many heuristics have been proposed [26, 32, 17, 31, 30, 33]. The problem is polynomial in some specific cases of localized failures: when SRLGs have span 1 (i.e. an edge can be affected by only one SRLG, and the set of edges belonging to the same SRLG forms a connected component, see [7]), and in a specific case of SRLGs having the star property [9] in which a link can be affected by at most two risks and two risks affecting the same link form stars at different nodes (this result also follows from [7]).

1.2 Our results

We study the k-diverse routing problem when SRLGs have the star property and there are no restrictions on the number of risks per link nor on the number of links per risk. This case has been studied in [24] in which the authors claim that finding two SRLG-disjoint paths under the star property can be solved in polynomial time. In this paper, we establish the following results:

1. We demonstrate that the algorithm proposed in [24] is not correct; indeed we exhibit, in Section 3, counterexamples for which their algorithm concludes to the non existence of two SRLG-disjoint paths although two such paths exist.

2. We prove, in Section 4, that finding k SRLG-disjoint paths is in fact NP-complete even only for two paths.

3. On the positive side, we show in Section 5, that the k-diverse routing problem can be solved in polynomial time in particular subcases which are relevant in practice. Namely, we solve the problem in polynomial time when the maximum degree is at most 4 or when the input network is a directed acyclic graph. Moreover, we show that the problem is fixed-parameter tractable when parameterized by the number of colors in $\mathcal{C}$.

4. Finally, we consider the problem of finding the maximum number of SRLG-disjoint paths. We prove that, under the star property, the problem is hard to approximate within $O(|V|^{1-\varepsilon})$ for any $0 < \varepsilon < 1$, where V is the set of nodes in the network, and we give polynomial time algorithms for some of the above relevant subcases.

We give the notation used in this paper in Section 2.

2 Notations and problem statement

We model the network as an undirected connected graph $G = (V, E)$, where the vertices in V represent the nodes and the edges in E represent the links. We associate a color with each SRLG. Let us denote by $\mathcal{C}$ the set of all the colors. Then a network with SRLGs is modeled by a *colored graph* that is a triple $G_c = (V, E, \mathscr{C})$, where (V, E) is a graph and $\mathscr{C}$ is a *coloring* function, $\mathscr{C} : E \to 2^{\mathcal{C}}$, that assigns a subset of colors to each edge of E.

We denote by $E(c)$ the set of edges having color $c \in \mathcal{C}$, by $\mathscr{C}(e)$ the set of colors associated with edge $e \in E$, by $\text{CPE} = \max_{e\in E} |\mathscr{C}(e)|$ the *maximum number of colors per edge*, and by $\text{EPC} = \max_{c\in\mathcal{C}} |E(c)|$ the *maximum number of edges having the same color*. We assume that $\mathscr{C}(e) \neq \emptyset$ for each $e \in E$. Given a vertex v, $\Gamma(v)$ denotes the set of neighbors of v and $d(v) = |\Gamma(v)|$ its *degree*. A color is *incident* to v if it is assigned to an edge incident to v. The *colored degree* of v, denoted by $d_{\mathcal{C}}(v)$, is the number of colors incident to v. The *maximum degree* and the *maximum colored degree* of a graph are denoted by Δ and $\Delta_{\mathcal{C}}$, respectively.

We can now model the star property defined in the introduction as follows. A color $c \in \mathcal{C}$ is called a *star color* if all edges of $E(c)$ are incident to the same vertex. A colored graph has the *star property* if it has only star colors.

Given a colored graph G_c and two vertices s and t, an st-path is an alternating sequence of vertices and edges, beginning with s and ending with t, in which each edge is incident to the vertex immediately preceding it and to the vertex immediately following it. A path is denoted by the sequence of vertices and edges. We say that two paths P_1 and P_2 are *color-disjoint* if $(\cup_{e\in P_1}\mathscr{C}(e)) \cap (\cup_{e\in P_2}\mathscr{C}(e)) = \emptyset$, i.e. the edges of one path do not have any color in common with the edges of the other path.

The k-diverse routing problem defined in the introduction consists then in finding k color-disjoint paths and for every k can be formally formulated as follows:

Problem 1 (k-Diverse Colored st-Paths, k-`DCP`). *Given a colored graph G_c and two vertices s and t, are there k color-disjoint paths from s to t?*

In this paper we study the k-`DCP` problem where the colored graphs have the star property.

3 Counterexamples to the algorithm of Luo and Wang

Luo and Wang [24] proposed an algorithm to find a pair of color-disjoint paths with minimum total cost from a source s to a destination t in graphs with colors (SRLGs) satisfying the star property. The algorithm is an adaptation of the Bhandari's edge-disjoint shortest-pair of paths algorithm [2, Chapter 3.3, pages 46-68] (which itself is a variation of the Suurballe-Tarjan's algorithm [27, 28]) and is based on augmenting a shortest path P_a between s and t.

In what follows, we argue that the algorithm is incorrect, as there are at least two problems with it.

Counterexample 1

The first problem comes from the fact that the algorithm implies that the first and last edges of the shortest s-t path P_a should be contained necessarily in the pair of paths returned by the algorithm. However, if no edge incident to s (to t) is color-disjoint with the first (the last) edge of P_a, respectively, the algorithm will ignore the existence of 2 color-disjoint paths even if they exist.

The counterexample in Figure 2 illustrates this first problem: color c is shared between edges $\{s, v_0\}$ and $\{s, v_1\}$ and color $c' \neq c$ is shared between edges $\{s, v_0\}$ and $\{s, v_2\}$. All other unmarked edges have distinct colors different from c and c'. The shortest path from s to t is $P_a = \{s, v_0, t\}$. Applied on the graph of Figure 2, the algorithm described in [24, page 451, lines 9–10] does not find any edge to start and hence terminates concluding that there are no two color-disjoint paths. However two color-disjoint paths clearly exist, namely they are $P_1 = \{s, v_1, w_1, t\}$ and $P_2 = \{s, v_2, w_2, t\}$.

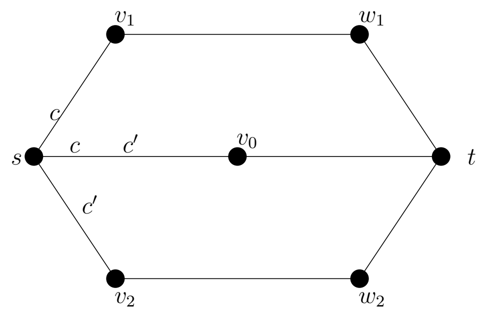


Figure 2: Example 1.

Counterexample 2

The second problem is that the algorithm only checks color-disjointness around nodes of P_a and never checks other nodes. It assumes implicitly that the only nodes that can be shared by the two color-disjoint paths are nodes belonging to P_a and ignores the existence of any other possibilities.

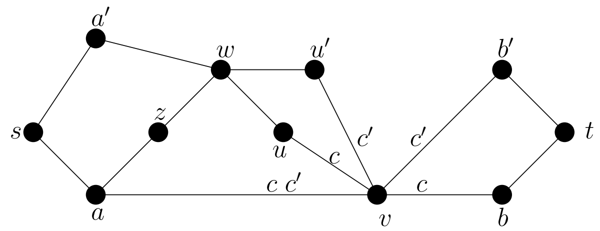
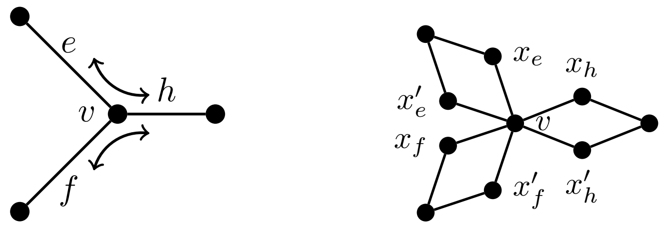


Figure 3: Example 2

Figure 3, illustrating the second problem, shows a counterexample to the algorithm in [24] that furthermore can give to the reader a flavor of the difficulty of the problem. In this figure, we have 2 specific colors c and $c' \neq c$ forming a star in v. All other unmarked edges have distinct colors different from c and c'. As vertex v is a cut-vertex any s-t path should contain v. Moreover $\{a, v\}$ cannot be used as it shares a color both with $\{v, b\}$ and $\{v, b'\}$. Therefore, to ensure the color-disjointness, one path should use the subpath u, v, b and the other one should use the subpath u', v, b'. We have two color-disjoint paths $P_1 = \{s, a, z, w, u, v, b, t\}$ and $P_2 = \{s, a', w, u', v, b', t\}$. However, the algorithm of [24] uses the shortest path $P_a = \{s, a, v, b, t\}$ and then performs a backwards phase which never finds w again. Then the algorithm terminates, missing the fact that there exist two color-disjoint paths. Note that the disjointness is not ensured, if there exists in w some common color for example on the edges (w, u) and (w, u'), showing that a local consideration around the shortest path is not sufficient. In fact, in the next section we will prove that the problem is NP-complete.



(a) v in G, $T(v) = \{\{e,h\},\{f,h\}\}$ (b) v in H

Edge	Colors
$\{v, x_e\}$	C_{ef}, $C_{ef'}$
$\{v, x'_e\}$	$C_{e'f}$, $C_{e'f'}$
$\{v, x_f\}$	C_{ef}, $C_{e'f}$
$\{v, x'_f\}$	$C_{ef'}$, $C_{e'f'}$
$\{v, x_h\}$	C_h,
$\{v, x'_h\}$	C'_h,

(c) Color assignment

Figure 4: Color assignment for vertices with degree 3.

4 NP-completeness

In this section we will prove that, even with the star property, the k-DCP problem is NP-complete for every constant $k \geq 2$. We use a reduction from the problem described below of finding a T-compatible path (or a path avoiding forbidden transitions), which was proven NP-complete in [29]. This again contradicts the supposed correctness of the polynomial algorithm of [24], unless $P = NP$.

Let $G = (V, E)$ be an undirected graph. A transition in $v \in V$ is a pair of edges incident to v. With each vertex v we associate a set $T(v)$ of admissible (or allowed) transitions in v. We call transition system the set $T = \{T(v) \mid v \in V\}$. Let $G = (V, E)$ be a graph and let T be a transition system. A path $P = \{v_0, e_1, v_1, \ldots, e_k, v_k\}$ in G, with $v_i \in V$, $e_i \in E$, is said to be T-compatible if, for every $1 \leq i \leq k-1$, the pair of edges $\{e_i, e_{i+1}\}$ is an admissible transition, i.e. $\{e_i, e_{i+1}\} \in T(v_i)$. We can now define the T-Compatible path problem.

Problem 2 (T-Compatible path, T-CP)**.** *Given a graph $G = (V, E)$, two vertices s and t in V, and a transition system T, does G contain a T-compatible path from s to t?*

It has been proven in [29] that problem T-CP is NP-complete and it remains NP-complete for the family $\mathcal{G}_4$ of simple graphs where vertices s and t have degree 3 and any other vertex has degree 3 or 4, and the set of transitions $T(v)$ is such that

- If $d(v) = 3$, $T(v)$ consists of two pairs of edges $\{e, h\}$ and $\{f, h\}$ where e, f and h are the 3 edges incident to v;
- If $d(v) = 4$, $T(v)$ consists of two pairs of distinct edges $\{e, f\}$ and $\{g, h\}$ where e, f, g and h are the 4 edges incident to v.

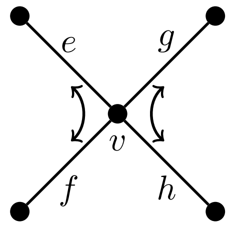
Theorem 1. *The k-DCP problem is NP-complete for any fixed constant $k \geq 2$, even if all the following properties hold:*

- *the star property;*
- *the maximum degree Δ is fixed with $\Delta \geq \max\{8, k\}$;*
- CPE, EPC *and $\Delta_{\mathcal{C}}$ are fixed with either [*CPE ≥ 4, EPC ≥ 2, *and* $\Delta_{\mathcal{C}} \geq \max\{16, k\}$*] or [*CPE ≥ 2, EPC ≥ 4 *and* $\Delta_{\mathcal{C}} \geq \max\{4, k\}$*].*

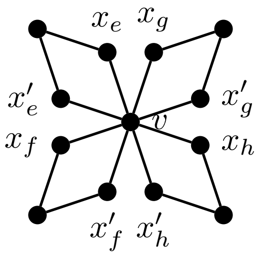
Proof. We first prove the statement for $k = 2$ and then extend it for any fixed $k \geq 3$.

It is easy to see that the problem is in NP since, given two paths we just have to check whether they are color-disjoint.

Given an instance (G, s, t, T) of the T-CP problem with G in the family $\mathcal{G}_4$, we define an instance of 2-DCP as follows. We associate with G a colored graph $H = (V_H, E_H, \mathscr{C})$ where:



(a) v in G, $T(v) = \{e, f\} \cup \{g, h\}$



(b) v in H

Edge	Colors
$\{v, x_e\}$	C_{eg}, $C_{eg'}$, C_{eh}, $C_{eh'}$
$\{v, x'_e\}$	$C_{e'g}$, $C_{e'g'}$, $C_{e'h}$, $C_{e'h'}$
$\{v, x_f\}$	C_{fg}, $C_{fg'}$, C_{fh}, $C_{fh'}$
$\{v, x'_f\}$	$C_{f'g}$, $C_{f'g'}$, $C_{f'h}$, $C_{f'h'}$
$\{v, x_g\}$	C_{eg}, $C_{e'g}$, C_{fg}, $C_{f'g}$
$\{v, x'_g\}$	$C_{eg'}$, $C_{e'g'}$, $C_{fg'}$, $C_{f'g'}$
$\{v, x_h\}$	C_{eh}, $C_{e'h}$, C_{fh}, $C_{f'h}$
$\{v, x'_h\}$	$C_{eh'}$, $C_{e'h'}$, $C_{fh'}$, $C_{f'h'}$

(c) Color assignment

Figure 5: Color assignment for vertices with degree 4.

- For each node in G, we associate a node in H;
- For each edge $e = \{u, v\}$ in G, we associate in H two nodes x_e and x'_e and two paths of length 2: $\{u, x_e, v\}$ and $\{u, x'_e, v\}$. H has then $|V(G)| + 2|E(G)|$ vertices and $4|E(G)|$ edges.
- We assign the colors to edges incident to a vertex v in H as follows:
  - Distinct new colors are assigned to the edges incident to t.
  - For each pair of edges e and f incident to s in G, such that $e \neq f$, we will use 4 colors $C_{ef}, C_{ef'}, C_{e'f}$ and $C_{e'f'}$. We assign colors C_{ef} and $C_{ef'}$ to the edge $\{s, x_e\}$; colors $C_{e'f}$ and $C_{e'f'}$ to the edge $\{s, x'_e\}$, colors C_{ef} and $C_{e'f}$ to the edge $\{s, x_f\}$ and colors $C_{ef'}$ and $C_{e'f'}$ to the edge $\{s, x'_f\}$.
  - For each $v \neq s, t$, and for each pair of edges e and f incident to v in G such that $e \neq f$ and $\{e, f\}$ is not an admissible transition (i.e. $\{e, f\} \notin T(v)$), we assign colors C_{ef} and $C_{ef'}$ ($C_{e'f}$ and $C_{e'f'}$) to the edge $\{v, x_e\}$ ($\{v, x'_e\}$), and colors C_{ef} and $C_{e'f}$ ($C_{ef'}$ and $C_{e'f'}$) to the edge $\{v, x_f\}$ ($\{v, x'_f\}$), respectively. As each vertex has either degree 3 or 4, two cases can occur:
    (i) If $d(v) = 3$, let e, f and h be the 3 edges incident to v and let $T(v) = \{\{e, h\}, \{f, h\}\}$, then the colors are assigned as described in Figure 4.
    (ii) If $d(v) = 4$, let e, f, g and h be the 4 edges incident to v and $T(v) = \{\{e, f\}, \{g, h\}\}$, then the colors are assigned as described in Figure 5.

The transformation is polynomial time computable and the star property holds. Moreover, note that each edge has at most 4 colors, each color is associated with two edges, the degree of each vertex is at most 8 and the color degree is at most 16. It follows that CPE ≤ 4, EPC ≤ 2, $\Delta \leq 8$, and $\Delta_{\mathcal{C}} \leq 16$.

To prove the theorem, we will use the following properties.

Property 1. *Given an edge e incident to s in G, the edge $\{s, x_e\}$ in H shares a color with all the other edges incident to s, except $\{s, x'_e\}$. In other words, the only pair of edges incident to s having no color in common are of the form $\{\{s, x_e\}, \{s, x'_e\}\}$ for some e.*

Property 2. *If $v \neq s, t$, $d(v) = 3$ and $T(v) = \{\{e, h\}, \{f, h\}\}$, then two edges incident to v share a color if and only if one is $\{v, x_e\}$ or $\{v, x'_e\}$ and the other is $\{v, x_f\}$ or $\{v, x'_f\}$.*

Property 3. *If $v \neq s, t$, $d(v) = 4$ and $T(v) = \{\{e, f\}, \{g, h\}\}$, then two edges incident to v share a color if and only if one is $\{v, x_e\}$, $\{v, x'_e\}$, $\{v, x_f\}$ or $\{v, x'_f\}$ and the other is $\{v, x_g\}$, $\{v, x'_g\}$, $\{v, x_h\}$ or $\{v, x'_h\}$.*

In other words, two edges incident to a node v are color-disjoint if and only if they correspond to an admissible transition of v or are of the form $\{v, x_e\}$ and $\{v, x'_e\}$.

We first show that if there exists a T-compatible path in G, then there exist two color-disjoint paths in H. Let $P = \{s, e_1, v_1, \ldots, e_p, v_p, e_{p+1}, t\}$ be a T-compatible path from s to t in G. Then $Q = \{s \equiv v_0, x_{e_1}, v_1, \ldots, x_{e_p}, v_p, x_{e_{p+1}}, t\}$ and $Q' = \{s \equiv v_0, x'_{e_1}, v_1, \ldots, x'_{e_p}, v_p, x'_{e_{p+1}}, t\}$ are two color-disjoint paths in H. In particular, by Properties 1, 2, and 3, any edge $\{v_i, x_{e_{i+1}}\}$ ($\{x_{e_i}, v_i\}$) has no color in common with $\{v_i, x'_{e_{i+1}}\}$ ($\{x'_{e_i}, v_i\}$), respectively, for each $i = 0, 1, \ldots, p$.

Conversely, we now show that if there exist two color-disjoint paths in H, then there exists a T-compatible path in G. Let the two color-disjoint paths in H be $Q = \{s, x_1, v_1, \ldots, x_p, v_p, x_{p+1}, t\}$ and $Q' = \{s, y_1, u_1, \ldots, y_{p'}, u_{p'}, y_{p'+1}, t\}$. We prove by induction on $i \in \{1, \ldots, p+1\}$, that $\{x_i, y_i\} = \{x_{e_i}, x'_{e_i}\}$, $v_i = u_i$ and $p = p'$.

For $i = 1$, by Property 1, $\{s, x_1\}$ and $\{s, y_1\}$ have no color in common only if $\{x_1, y_1\} = \{x_e, x'_e\}$ for an edge e incident to s and then $v_1 = u_1$.

Let us suppose that the statement is true until $i = l$; we will prove it for $i = l + 1$. Let the two edges entering $u_l = v_l$ used by Q and Q' be $\{x_{e_l}, v_l\}$ and $\{x'_{e_l}, v_l\}$.

If $d(v_l) = 3$, we distinguish two cases:

- e_l belongs to only one admissible pair of $T(v_l)$ say $\{e_l, h_l\}$ and the paths Q and Q', being color-disjoint, can only use the edges $\{v_l, x_{h_l}\}$ and $\{v_l, x'_{h_l}\}$.
- e_l belongs to two admissible pairs in $T(v_l)$, $\{e_l, f_l\}$ and $\{e_l, h_l\}$. If one path uses the edge $\{v_l, x_{h_l}\}$ ($\{v_l, x'_{h_l}\}$) the other path cannot use the edge $\{v_l, x_{f_l}\}$ or $\{v_l, x'_{f_l}\}$ by Property 2, it has then to use edge $\{v_l, x'_{h_l}\}$ ($\{v_l, x_{h_l}\}$), respectively.

Therefore, in both cases $\{x_{l+1}, y_{l+1}\} = \{x_{h_l}, x'_{h_l}\}$ and $v_{l+1} = u_{l+1}$.

If $d(v_l) = 4$, by Property 3, the only possibility as Q and Q' are color-disjoint is that they use the edges $\{v_l, x_{e_{l+1}}\}$ and $\{v_l, x'_{e_{l+1}}\}$, respectively, where $\{e_l, e_{l+1}\} \in T(v_l)$ and so the statement is true for $i = l + 1$.

It follows that the path $P = \{s, e_1, v_1, \ldots, e_p, v_p, e_{p+1}, t\}$ satisfies $\{e_i, e_{i+1}\} \in T(v_i)$ for every $i \in \{1, \ldots, p\}$ and then it is T-compatible.

To show that the problem remains NP-complete even for fixed CPE ≥ 2, EPC ≥ 4 and $\Delta_{\mathcal{C}} \geq 4$, it is enough to modify the above transformation by using a different color assignment. In detail, the color assignment differs from the one given above as follows:

- Edges incident to vertex s (which has degree 3) have the color assignment reported in Table 1a;
- Edges incident to vertices with degree 4 in G have the color assignment reported in Table 1b;
- The other vertices (i.e. t and those with degree 3 in G) keep the same color assignment as before.

The above proof works with this color assignment with slight changes. Indeed in Property 3 edge $\{v, x_e\}$ shares colors with edge $\{v, x_f\}$. But the proof is still valid as when $d(v_l) = 4$, if Q (Q') uses the edge $\{v_l, x_{e_l}\}$ ($\{v_l, x'_{e_l}\}$), then Q (Q') uses the edge $\{v_l, x_{e_{l+1}}\}$ ($\{v_l, x'_{e_{l+1}}\}$), respectively, where $\{e_l, e_{l+1}\} \in T(v_l)$. It follows that 2-DCP is NP-hard even for fixed CPE ≥ 2, EPC ≥ 4 and $\Delta_{\mathcal{C}} \geq 4$.

Edge	Colors
$\{s, x_e\}$	C_1, C_2
$\{s, x'_e\}$	C_3, C_4
$\{s, x_f\}$	C_1, C_3
$\{s, x'_f\}$	C_2, C_4
$\{s, x_h\}$	C_1, C_4
$\{s, x'_h\}$	C_2, C_3

(a) Color assignment for vertex s.

Edge	Colors
$\{v, x_e\}$	C_1, C_2
$\{v, x'_e\}$	C_3, C_4
$\{v, x_f\}$	C_1, C_2
$\{v, x'_f\}$	C_3, C_4
$\{v, x_g\}$	C_1, C_3
$\{v, x'_g\}$	C_2, C_4
$\{v, x_h\}$	C_1, C_3
$\{v, x'_h\}$	C_2, C_4

(b) Color assignments for vertices with degree 4.

Table 1: Color assignments for vertex s (Table 1a) and for vertices with degree 4 (Table 1b) when CPE ≥ 2 and EPC ≥ 4.

We can extend the proof to the case where $k \geq 3$ in various ways. In a first version we added $k-2$ paths of length 2 from s to t, $P_i = \{s, w_i, t\}$ for $i = 3, 4, \ldots k$, with a new color assigned to each edge $\{s, w_i\}$. The following construction which gives better results was suggested by one referee; we modify $H = (V_H, E_H, \mathscr{C})$ as follows:

- We introduce two additional vertices s' and t'.
- We add $k-2$ paths of length 2 from s' to t', $P_i = \{s', w_i, t'\}$ for $i = 3, 4, \ldots k$, with a new color assigned to each edge $\{s', w_i\}$ and $\{t', w_i\}$. These paths are pairwise color-disjoint.
- We add two paths of length 2 from s to s', $A_i = \{s', a_i, s\}$ for $i = 1, 2$, with a new color assigned to each edge $\{s, a_i\}$ and $\{s', a_i\}$.
- We add two paths of length 2 from t to t', $B_i = \{t, b_i, t'\}$ for $i = 1, 2$, with a new color assigned to each edge $\{t, b_i\}$ and $\{t', b_i\}$.

Finding k color-disjoint paths from s' to t' in this new graph is equivalent to finding 2 color-disjoint paths from s to t . Moreover, this assignment does not change CPE and EPC,and Δ and $\Delta_{\mathcal{C}}$ are either the same or equal to k. $\square$

5 Polynomial cases

In this section we give polynomial time algorithms for k-DCP for some important special cases. In detail, we solve k-DCP for the cases where the number of colors is bounded by a constant (i.e. $|\mathcal{C}| = O(1)$), for some cases where the maximum degree of the graph is strictly smaller than 5, and for the cases where the input graph is a Directed Acyclic Graph (DAG). All the given algorithms work only when the star property hold, but the one for the case when $|\mathcal{C}| = O(1)$ which works for any possible color assignment.

5.1 Bounded number of colors

In this section, we give an algorithm to find k color-disjoint paths in the special case where the number $|\mathcal{C}|$ of colors in the network is bounded by a constant, i.e. $|\mathcal{C}| = O(1)$. We observe that such an algorithm works for every graph topology and even if the star property does not hold.

We will reduce our problem to the Set Packing problem.

Problem 3 (Set Packing)**.** *Given a set X, a collection $\mathcal{S}$ of subsets of X and an integer k, is there a collection of disjoint sets $\mathcal{S}' \subseteq \mathcal{S}$ such that $|\mathcal{S}'| = k$?*

The Set Packing problem is known to be NP-hard [16, Problem SP3, page 221] but is polynomial-time solvable when the size of X is bounded [3].

A subset $A \subseteq \mathcal{C}$ of colors will be called *realizable*, if the subgraph G_A induced by the edges whose colors are *all* in A (i.e. edges e such that $\mathscr{C}(e) \subseteq A$) contains at least one path from s to t. Note that such a path uses only colors of A.

The idea of the algorithm is to enumerate all the realizable subsets of $\mathcal{C}$ and then find k disjoint realizable subsets by using an exact algorithm for the Set Packing. As the size of $\mathcal{C}$ is constant, the computational time required by such algorithm is polynomial.

The details of the algorithm along with its correctness and complexity are given in the next theorem.

Theorem 2. *The k-*`DCP` *problem is FPT when parameterized by the number of colors $|\mathcal{C}|$. In particular, there exists an algorithm for solving the k-*`DCP` *problem in time $O(f(|\mathcal{C}|)(|V| + |E|))$, where f is a function depending solely on $|\mathcal{C}|$.*

Proof. Let $X = \mathcal{C}$ and let $\mathcal{S}$ be the family of realizable subsets of colors. Then there exists a collection of k disjoint sets $\mathcal{S}' \subseteq \mathcal{S}$ if and only if there exist k color-disjoint paths from s to t. Indeed, to each subset A' of $\mathcal{S}'$ is associated a path using uniquely colors of A' (as A' is realizable) and two disjoint subsets correspond to two color-disjoint paths.

Determining if a subset of colors is realizable requires $O(|V| + |E|)$ computational time. Furthermore, it is known that there exist polynomial time algorithms to solve the Set Packing problem when the size of X is bounded. For instance, the exact algorithm proposed in [3] has time complexity $O(|\mathcal{S}|2^{|X|}|X|^{O(1)})$. As $|X| = |\mathcal{C}|$ and $|\mathcal{S}| \leq 2^{|\mathcal{C}|}$, we deduce that finding k color-disjoint paths requires $O(2^{2|\mathcal{C}|}|\mathcal{C}|^{O(1)}(|V| + |E|))$ overall time, and so that the k-`DCP` problem is FPT when parameterized by the number of colors $|\mathcal{C}|$. □

5.2 Bounded degree

In this section, we assume that the star property holds and that $|\mathcal{C}|$ is unbounded and we give algorithms for finding k color-disjoint paths when $\Delta < 4$ and for finding 2 color-disjoint paths when $\Delta = 4$. First, note that the maximum number of color-disjoint paths in a graph is upper bounded by Δ.

If $\Delta \leq 2$ the problem is trivial as the graph is either a path or a cycle. In the first case, there always exists only one path from s to t. In the second case, the only vertices where the two possible paths can share colors are s and t and hence it is enough to check if the two edges incident to s (and t) are color-disjoint.

If $\Delta \leq 3$, observe that if two paths share an internal vertex of degree 3, they necessarily share also an edge and hence all the colors of that edge. Consequently, they cannot be color-disjoint. Furthermore, if two paths are color-disjoint the colors of their first edges should be disjoint and also the colors of their last edges should be disjoint.

If $\Delta = 3$ and $k = 3$, there are 3 color-disjoint paths if and only if G has 3 vertex-disjoint paths between s and t and the 3 first edges of these paths have disjoint colors and also the 3 last edges. That can be checked in $O(|V| + |E|)$ time: constant time for checking the color disjointness of the 3 first (last) edges, and $O(|V| + |E|)$ time for checking the existence of 3 vertex-disjoint paths between s and t (see [15]).

If $k = 2$ and $\Delta = 3$ or 4, we give an algorithm in the proof of the following theorem:

Algorithm 1: Solving 2-DCP when $\Delta = 3, 4$.

1 **foreach** *admissible graph* $G(s_i, s_j, t_{i'}, t_{j'})$ **do**

2   **if** *there exist 2 vertex-disjoint paths from* s *to* t *in* $G(s_i, s_j, t_{i'}, t_{j'})$ **then**

3   There exist two color-disjoint paths from s to t in G;

4   **else**

5   **if** *all the cut-vertices that separate* s *from* t *in* $G(s_i, s_j, t_{i'}, t_{j'})$ *have degree 4 and are not incident to bridges* **then**

6   **foreach** *cut vertex* v *that separates* s *from* t *in* $G(s_i, s_j, t_{i'}, t_{j'})$ **do**

7   Let e and f be the edges incident to v in the connected component containing s, and let e' and f' be the incident edges in the connected component containing t;

8   **if** ***not***
$$\left(\begin{array}{cc} & \mathscr{C}(e) \cap \mathscr{C}(f) = \emptyset \textit{ and } \mathscr{C}(e') \cap \mathscr{C}(f') = \emptyset \\ \textit{and} & \left[\begin{array}{ll} & [\mathscr{C}(e) \cap \mathscr{C}(e') = \emptyset \textit{ and } \mathscr{C}(f) \cap \mathscr{C}(f') = \emptyset] \\ \textit{or} & [\mathscr{C}(e) \cap \mathscr{C}(f') = \emptyset \textit{ and } \mathscr{C}(f) \cap \mathscr{C}(e') = \emptyset] \end{array}\right] \end{array}\right)$$
                then

9   No 2 color-disjoint paths from s to t exist in G;

10   There exist two color-disjoint paths from s to t in G;

11 No 2 color-disjoint paths from s to t exist in G.

Theorem 3. *Algorithm 1 solves 2-*DCP *in graphs with the star property and* $\Delta = 3$ *or* 4 *in time* $O(|V| + |E|)$.

Proof. We say that a pair $\{s_i, s_j\}$ of neighbors of s in G is admissible, if the edges joining s to them have disjoint colors, i.e. $\mathscr{C}(\{s, s_i\}) \cap \mathscr{C}(\{s, s_j\}) = \emptyset$ and similarly we say that $\{t_{i'}, t_{j'}\}$ is an admissible pair of neighbors of t if the edges joining them to t have disjoint colors. Then, with each admissible pair $\{s_i, s_j\}$ and each admissible pair $\{t_{i'}, t_{j'}\}$, we associate the admissible graph $G(s_i, s_j, t_{i'}, t_{j'})$ obtained from G by deleting the edges $\{s, s_\ell\}$ with $\ell \neq i, j$ and the edges $\{t_{\ell'}, t\}$ with $\ell' \neq i', j'$.

We solve 2-DCP when $\Delta = 3, 4$ by using Algorithm 1 whose correctness is given in what follows.

Note first that since all colors satisfy the star property, they are localized around vertices and the color-disjointness of two paths can be ensured by the color-disjointness around their shared vertices.

By definition, if there exist two vertex-disjoint paths from s to t in $G(s_i, s_j, t_{i'}, t_{j'})$ the first edges and last edges of such paths have disjoint colors and so we conclude that there are 2 color-disjoint paths (lines $2, 3$).

Otherwise, if there exists a cut vertex (i.e., a vertex which removal disconnects s from t and hence should be included in any path from s to t) v of degree 3, we cannot have color-disjoint paths containing this vertex. That is in particular the case when $\Delta = 3$. If there exists a cut-vertex v of degree 4 that is incident to a bridge, then v belongs to at most one 2-connected component which is not a bridge. In this case, there are no two color-disjoint paths from s to t. So, let us now assume that $\Delta = 4$, all the cut vertices are of degree 4 and, each cut vertex is incident to two 2-connected components which are not bridges. Between every two cut vertices, the two paths use vertex-disjoint subpaths. If at the cut vertex v one path uses edges e and e' (e and f'), the other path uses necessarily f and f' (f and e'), respectively, and the conditions on colors are necessary and sufficient for the color disjointness of the paths at v (the center of the colors used in v).

Since we have at most 6 admissible pairs of neighbors of s (of t), respectively, we have at most 36 graphs to consider. For each graph we have to check if it is 2-connected (that can be done in time $O(|V| + |E|)$ [19]) and if it is not 2-connected to satisfy coloring conditions at each cut vertex (all of the cut-vertices can be determined in linear time using the algorithm for finding the biconnected components of a graph in [19]), which can be done in constant time for a given vertex and so overall in time $O(|V|)$. □

Note that Algorithm 1 cannot be extended in a straightforward manner neither to find 3 or 4 color-disjoint paths on a graphs with $\Delta = 4$, nor to the case of $\Delta = 5, 6, 7$ in polynomial time. In fact, for these cases, while the number of admissible graphs stays bounded and finding vertex-disjoint path segments stays polynomial, the number of possible ways to cross the cuts explodes exponentially.

5.3 Directed acyclic graphs

In this section, we propose an algorithm for finding k color-disjoint paths in a colored directed acyclic graph (DAG) with the star property. The definitions given for undirected colored graphs can be easily extended to colored DAGs by assigning an acyclic orientation to the edges of the graph. As each color is a star color we can associate with each color c its center v defined as the common vertex to all arcs with color c. If the color has only one occurrence we choose arbitrarily as associated center one of the end vertices of the arc containing this color. We will say that the color c is centered in v.

The algorithm given in the proof of the next theorem uses ideas of [6], in particular that of layered directed graph and a construction similar to that used to find a polynomial time algorithm for disjoint paths with forbidden pairs (Theorem 6 of [6]).

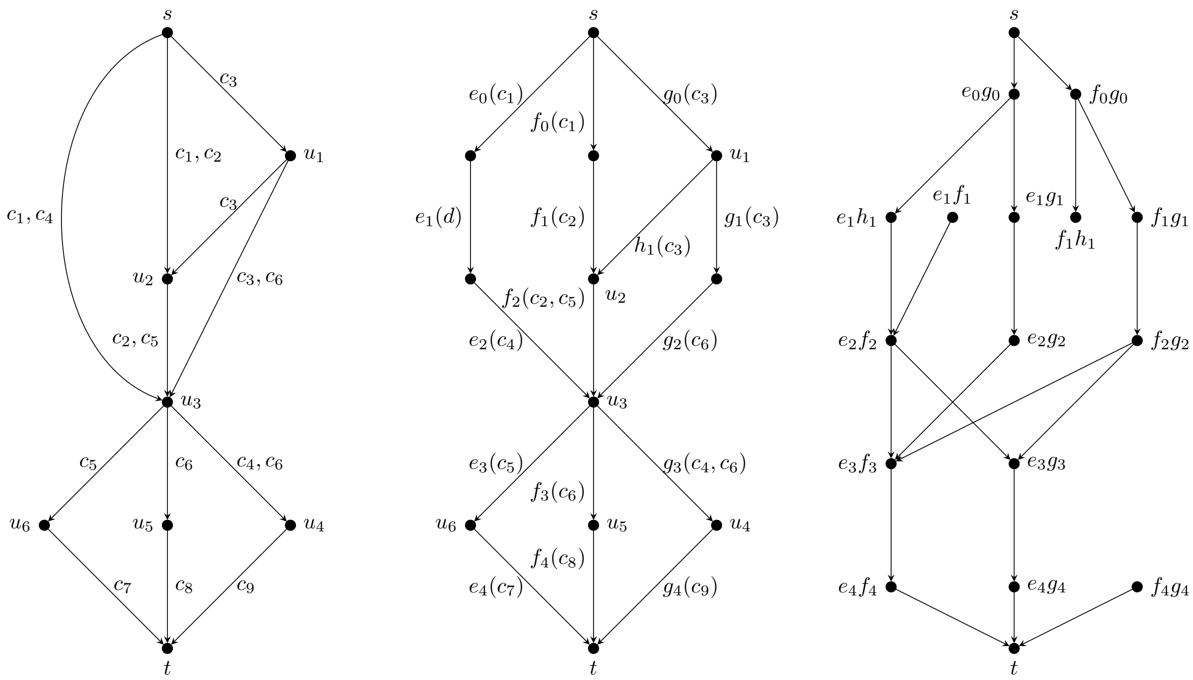
Definition 1 (Layered directed graph). *A directed graph $G = (V, E)$ is layered if there is a layering function $l : V \to [0, 1, \dots, (|V| - 1)]$ such that for every arc $(u, v) \in E$, $l(v) = l(u) + 1$. We say that vertex u is in layer $l(u)$ and arc (u, v) is in layer $l(u)$. Layered directed graphs are acyclic.*

In Theorem 4, we present an algorithm for solving the k-DCP problem in a DAG with the star property in time $O(\text{CPE}^2|V||E|^{2k})$. This algorithm is therefore polynomial only when k is a fixed constant.

We will then show in Section 6 that the problem is $W[1]$-hard and therefore it is not possible to find an FPT algorithm (i.e. having time complexity $O(f(k) \cdot poly(|V| + |E|))$, for any function f), unless $FPT = W[1]$.

Theorem 4. *There exists an algorithm that solves k-DCP in a DAG with the star property in time$O(\text{CPE}^2|V||E|^{2k})$.*

Proof. Let D be a multicolored DAG and let s and t be two given vertices. As we want to find (in polynomial time) directed paths from s to t, we can delete the vertices not on a directed path from s to t, and so we suppose in what follows that D is this reduced DAG. Now s is the unique vertex with no predecessor and t the unique vertex with no successor. The algorithm that finds k color-disjoint paths from s to t in D uses two transformations:



(a) Example of muticolored DAG D

(b) LD obtained after the layering of D

(c) The graph H associated with LD

Figure 6: Transformations for the DAG.

Transformation 1 We first associate with a multicolored DAG D a multicolored layered DAG LD as follows. We denote by $\Gamma^-(v)$ the set of vertices preceding v, i.e; vertices u such that $(u, v) \in E$. We compute the function $l : V \to \mathbb{N}$ defined as follows:

$$l(v) = \begin{cases} 0 & \text{when } v = s, \\ 1 + \max_{u \in \Gamma^-(v)} l(u) & \text{otherwise.} \end{cases}$$

In such a level function, t has the maximum value as there is a directed path from any vertex to t in the reduced DAG. In the example of Figure 6a, we have $l(u_1) = 1$, $l(u_2) = 2$, $l(u_3) = 3$, $l(u_4) = l(u_5) = l(u_6) = 4$, and $l(t) = 5$.

Now we replace every arc (u, v), such that $l(v) > l(u) + 1$, with a directed path P_{uv} from u to v of length $l(v) - l(u)$ (thus possibly adding new vertices and arcs). We assign to the first arc of the directed path P_{uv} the colors of the arc (u, v) centered in u (or a new color if there are no colors centered in u) and to the last arc of the directed path P_{uv} the colors of the arc (u, v) centered in v (or a new color if there are no colors centered in v) and to the intermediate arcs, if any, new distinct colors. The resulting layered DAG LD is such that there exist k color-disjoint paths in the DAG D from s to t if and only if there exist k color-disjoint paths in LD from s to t. In Figure 6b, we indicate the layered DAG LD obtained from the DAG D of Figure 6a. We have given a name to each arc with a lower index indicating the level of the arc; we also indicate inside parentheses the colors attributed to each arc. For example the arc (s, u_3) which had colors c_1, c_4 has been replaced by a path with 3 arcs: e_0 at level 0 which gets the color c_1 (centered at s), e_1 at level 1 which gets a new color d and e_2 at level 2 which gets the color c_4 (centered at u_3).

Therefore, in what follows we consider a layered DAG LD with two specific vertices s and t.

Transformation 2 We use in this transformation ideas similar to that used in [6] to solve the problem of finding a pair of vertex-disjoint paths with forbidden pairs of edges. Here instead of vertex-disjointness we seek edge-disjointness and the forbidden pairs of edges are forbidden pairs of subpaths sharing a color.

With LD we will associate a directed graph H with two specific vertices s and t, such that there exist k color-disjoint directed paths in LD from s to t if and only if there exists a directed path from s to t in H. For the ease of presentation, we first give the transformation for $k = 2$.

There is a vertex in H for every pair $\{e_i, f_i\}$ of arcs in LD at the same layer i, with $0 \leq i \leq l(t) - 1$, such that $\mathscr{C}(e_i) \cap \mathscr{C}(f_i) = \emptyset$. We also add to H two vertices s and t. Now we join, by an arc in H, s to all the vertices (pairs) $\{e_0, f_0\}$. Similarly we join every vertex $\{e_{l(t)-1}, f_{l(t)-1}\}$ in H by an arc to t. Finally, for $0 \leq i \leq l(t) - 2$, we join in H each vertex $\{e_i, f_i\}$ to a vertex $\{e_{i+1}, f_{i+1}\}$ if in LD we have the following properties:

1. the terminal vertex u_i (v_i) of e_i (f_i) is the initial vertex of e_{i+1} (f_{i+1}), respectively, and
2. either $u_i \neq v_i$
3. or $u_i = v_i$ and the set of colors of $\mathscr{C}(e_i) \cup \mathscr{C}(e_{i+1})$ is disjoint from the set of colors of $\mathscr{C}(f_i) \cup \mathscr{C}(f_{i+1})$
4. or $u_i = v_i$ and the set of colors of $\mathscr{C}(e_i) \cup \mathscr{C}(f_{i+1})$ is disjoint from the set of colors of $\mathscr{C}(f_i) \cup \mathscr{C}(e_{i+1})$.

Figure 6c indicates the graph H obtained from the layered DAG LD of Figure 6b. For example, we have three vertices corresponding to pairs of arcs of layer 2 of LD: e_2f_2, e_2g_2 and f_2g_2 but only two vertices corresponding to pairs of arcs of layer 3: e_3f_3 and e_3g_3. Vertex e_2g_2 is connected to vertex e_3g_3 as condition 3 is fulfilled but it is not connected to f_3g_3 as none of the conditions 3 and 4 is fulfilled.

The existence of two disjoint colored directed paths in LD named $P = (s, e_0, u_0, e_1, \ldots, e_{l(t)-2}, u_{l(t)-1}, e_{l(t)-1}, t)$ and $Q = (s, f_0, v_0, f_1, \ldots, f_{l(t)-2}, v_{l(t)-1}, f_{l(t)-1}, t)$ implies the existence of a directed path from s to t namely $PQ = (s, \{e_0, f_0\}, \{e_1, f_1\}, \ldots, \{e_{l(t)-1}, f_{l(t)-1}\}, t)$ in H.

Conversely, let W be a path in H written in the form $W = (s, w_0, w_1, \ldots, w_{l(t)-1}, t)$ where w_i corresponds to the pair $\{e_i, f_i\}$ and w_{i+1} to the the pair $\{e_{i+}, f_{i+1}\}$ such that the set of colors of $\mathscr{C}(e_i) \cup \mathscr{C}(e_{i+1})$ is disjoint from the set of colors of $\mathscr{C}(f_i) \cup \mathscr{C}(f_{i+1})$; such ordering is possible since one of the color conditions is fulfilled. Then, the two directed paths $P = (s, e_0, u_0, e_1, \ldots, e_{l(t)-2}, u_{l(t)-1}, e_{l(t)-1}, t)$ and $Q = (s, f_0, v_0, f_1, \ldots, f_{l(t)-2}, v_{l(t)-1}, f_{l(t)-1}, t)$ are color-disjoint. In the example of Figure 6c, H has many directed paths from s to t. For example with the directed path $P = (s, \{e_0, g_0\}, \{e_1, h_1\}, \{e_2, f_2\}, \{e_3, g_3\}, \{e_4, g_4\}, t)$, the two color-disjoint directed paths $P_1 = (s, e_0, e_1, e_2, g_3, g_4, t)$ and $P_2 = (s, g_0, h_1, f_2, e_3, e_4, t)$ in LD and the two color-disjoint directed paths (s, u_3, u_4, t) and $(s, u_1, u_2, u_3, u_6, t)$ in D are associated.

The algorithm can be generalized to find k color-disjoint paths from s to t in a DAG D, for any $k \geq 2$. We first transform D to a layered graph LD as before. Then, in the second transformation, instead of having a vertex for every pair of arcs of the same layer, we create a vertex for every k-tuple of arcs $\{e_i^1, e_i^2, \ldots, e_i^k\}$ at the same layer i, such that the $\mathscr{C}(e_i^j)$, for $j = 1, \ldots, k$, are disjoint. Then an arc is added from node $\{e_i^1, e_i^2, \ldots, e_i^k\}$ to node $\{e_{i+1}^1, e_{i+1}^2, \ldots, e_{i+1}^k\}$ if there exists an ordering of the e_i^j and of the e_{i+1}^j such that the terminal vertex of e_i^j is the initial vertex of e_{i+1}^j and the sets $\mathscr{C}(e_i^j) \cup \mathscr{C}(e_{i+1}^j)$ are pairwise disjoint.

To decide if a k-tuple $\{e_i^1, e_i^2, \ldots, e_i^k\}$ is a vertex of H, we need to check the color-disjointness of $\mathscr{C}(e_i^j)$, for $j = 1, \ldots, k$. This can be done in at most $\frac{k(k-1)}{2}\text{CPE}^2$ steps. Deciding on the existence of an edge between two vertices of H can be done in $k!k(k-1)\text{CPE}^2$ ($O(\text{CPE}^2)$) time; indeed we can choose an ordering of $\{e_i^1, e_i^2, \ldots, e_i^k\}$ and for each of the $k!$ possible orderings of $\{e_{i+1}^1, e_{i+1}^2, \ldots, e_{i+1}^k\}$, we check whether the sets $\mathscr{C}(e_i^j) \cup \mathscr{C}(e_{i+1}^j)$ are pairwise disjoint in at most $k(k-1)\text{CPE}^2$ steps. Finally, as each arc in D is replaced in LD by a path containing at most one arc of each layer, the number of arcs at a given layer in LD is at most $|E|$. So, the graph H has at most $l(t)|E|^k$ vertices and $l(t)|E|^{2k}$ edges. Therefore, we get the complexity of the theorem as $l(t) \leq |V|$.

□

Remark 1. *The algorithm presented in the proof of Theorem 4 can be adapted to find a minimum cost pair of color-disjoint paths in an arc-weighted DAG by applying the following modifications.*

Let us consider a weight function on the arcs of a DAG D. We assign the original weight of the arc (u, v) to the first arc of the path replacing it in LD. Then, in H we assign to the edge joining s to $\{e_0, f_0\}$ the sum of the weights of e_0 and f_0, and to the edge joining $\{e_i, f_i\}$ to $\{e_{i+1}, f_{i+1}\}$ the sum of the weights of $\{e_{i+1}$ and $f_{i+1}\}$. With these modifications, the shortest path in H corresponds to the optimal pair of color-disjoint paths in D.

Remark 2. *We can also use the algorithm presented in the proof of Theorem 4 to find a pair of color-disjoint paths with the minimum total number of colors by applying the following modifications.*

Let $\mathscr{C}^-((u, v))$ ($\mathscr{C}^+((u, v))$) be the set of colors of arc (u, v) centered at u (v), respectively. In H, we assign to the arc from $\{e_i, f_i\}$ to $\{e_{i+1}, f_{i+1}\}$ a weight equal to $|\mathscr{C}^+(e_i) \cup \mathscr{C}^-(e_{i+1}) \cup \mathscr{C}^+(f_i) \cup \mathscr{C}^-(f_{i+1})|$, to the arc from s to $\{e_0, f_0\}$ a weight equal to $|\mathscr{C}^-(e_0) \cup \mathscr{C}^-(f_0)|$ and to the arc from $\{e_{l(t)-1}, f_{l(t)-1}\}$ to t a weight equal to $|\mathscr{C}^+(e_{l(t)-1}) \cup \mathscr{C}^+(f_{l(t)-1})|$. We have proven above that every directed path P from s to t in H corresponds to two color-disjoint directed paths P_1 and P_2 from s to t in the layered graph LD (and equivalently to two color-disjoint paths from s to t in D) and with the way we have defined the weights in H, the weight of P is equal to the number of colors used by P_1 and P_2. The shortest path in the weighted graph H will then correspond to the pair of color-disjoint paths with the minimum number of colors.

6 Maximum number of color-disjoint paths

In this section we reformulate the problem of finding SRLG-disjoint paths as an optimization problem where we aim at finding the maximum number of color-disjoint paths:

Problem 4 (Max Diverse Colored st-Paths, `MDCP`)**.** *Given a colored graph G_c and two vertices s and t, find the maximum number of color-disjoint st-paths.*

In the next theorem we give complexity results for `MDCP` by using an approximation factor preserving reduction from Maximum Set Packing (`MSP`).

Definition 2 (Maximum Set Packing, `MSP`)**.** *Given a set X and a collection $\mathcal{S}$ of subsets of X, find the maximum cardinality set packing, i.e., a collection of disjoint sets $\mathcal{S}' \subseteq \mathcal{S}$ such that $|\mathcal{S}'|$ is maximized.*

It has been proven that problem `MSP` is equivalent to the problem of finding a maximum clique in a graph under a PTAS reduction where the number n of vertices in the graph corresponds to $|\mathcal{S}|$ [1]. In detail, approximation algorithms and inapproximability results (in terms of the number of vertices in the graph) carry over to the `MSP` problem. It is NP-hard to approximate the problem of finding a maximum clique within $O(n^{1-\varepsilon})$, for any $0 < \varepsilon < 1$ [20] and then, unless $P = NP$, `MSP` is not approximable within $O(|\mathcal{S}|^{1-\varepsilon})$, for any $0 < \varepsilon < 1$. Moreover, if the cardinality of all sets in $\mathcal{S}$ is upper bounded by a constant $c \geq 3$, then the problem is

APX-complete [23]. The next theorem gives inapproximability results for the MDCP problem. The proof uses an approximation-preserving reduction from MSP to MDCP where the vertices of V correspond to the elements of $\mathcal{S}$.

Theorem 5. *Unless $P = NP$,* MDCP *cannot be approximated within $O(|V|^{1-\varepsilon})$, for any $0 < \varepsilon < 1$, even if* EPC *is fixed,* EPC ≥ 2. *Moreover, it is APX-hard if* CPE *is fixed,* CPE ≥ 3. *These inapproximability results hold even in DAGs with the star property.*

Proof. Given an instance $I_{\mathtt{MSP}}$ of MSP over a set X and a collection $\mathcal{S}$, we define an instance $I_{\mathtt{MDCP}}$ of MDCP on a graph G_C as follows.

- for each element S_i of S, we associate a vertex v_{S_i};
- we add two vertices s and t and the edges $\{s, v_{S_i}\}$ and $\{v_{S_i}, t\}$, for each $S_i \in S$;

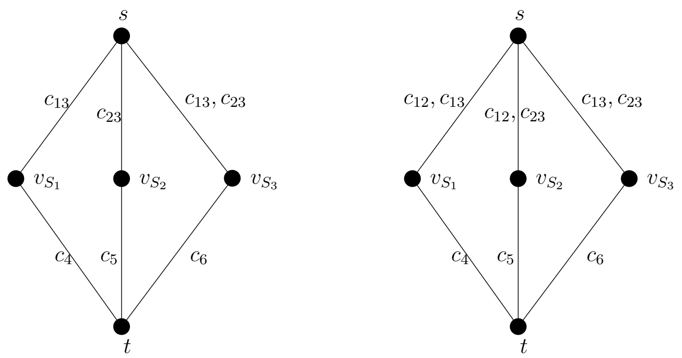


Figure 7: Examples where $\mathcal{S} = \{S_1 = \{a, b, c\}, S_2 = \{d, e, f\}, S_3 = \{a, b, d\}\}$ (left) and $\mathcal{S} = \{S_1 = \{a, b, d\}, S_2 = \{b, c, d\}, S_3 = \{b, e, f\}\}$ (right).

First Coloring

- for each $S_i, S_j \in \mathcal{S}$, such that $i \neq j$ and $S_i \cap S_j \neq \emptyset$, we add a new color c_{ij} and associate it with $\{s, v_{S_i}\}$ and $\{s, v_{S_j}\}$;
- for each edge not yet colored (in particular for each edge $\{v_{S_i}, t\}$) we put a new color.

See Figure 7 for two examples. In the left figure as $S_1 \cap S_3 \neq \emptyset$ and $S_2 \cap S_3 \neq \emptyset$ we have a color c_{13} on $\{s, v_{S_1}\}$ and $\{s, v_{S_3}\}$ and a color c_{23} on $\{s, v_{S_2}\}$ and $\{s, v_{S_3}\}$. On the right figure we have furthermore a color c_{12} on $\{s, v_{S_1}\}$ and $\{s, v_{S_2}\}$ as $S_1 \cap S_2 \neq \emptyset$. By definition each color is associated with at most two edges and hence EPC ≤ 2.

Second Coloring

We define a color c_x for each element $x \in X$ and, for each $S_i = \{x_1, \ldots, x_h\} \in C$, we associate the $|S_i|$ colors $c_{x_1}, \ldots, c_{x_h}$ with the edge $\{s, v_{S_i}\}$. For each edge not yet colored (in particular for each edge $\{v_{S_i}, t\}$ we put a new color. In this way, if the cardinality of all sets in $\mathcal{S}$ is upper bounded by a constant $c \geq 3$, then CPE $\leq c$.

In both cases the transformation is polynomial-time computable and the star property holds. Furthermore, the graph we obtain is the bipartite complete graph $K_{2,|\mathcal{S}|}$. This graph can be oriented into a DAG in a straightforward way.

Now we can associate to a family $\mathcal{S}' = \{S_1, S_2, \ldots, S_q\}$ of sets in $\mathcal{S}$, the set $\mathcal{P}$ of paths $P_j = (s, v_{S_j}, t)$, $j = 1, 2, \ldots, |\mathcal{S}'|$ of G_C and vice versa. Note that by construction two sets S_i and S_j are disjoint if and only if the corresponding paths P_i and P_j are color-disjoint.

Consider an optimal solution $\mathcal{S}'_{OPT}$ for MSP, with $\mathcal{S}'_{OPT} = \{S_1, S_2, \ldots, S_{|\mathcal{S}'_{OPT}|}\}$; the associated set $\mathcal{P}$ of paths $P_j = (s, v_{S_j}, t)$, for each $j = 1, 2, \ldots, |\mathcal{S}'_{OPT}|$ is a feasible solution for $I_{\texttt{MDCP}}$ with $|\mathcal{P}| = |\mathcal{S}'_{OPT}|$, and so,

$$OPT(I_{\texttt{MSP}}) \leq OPT(I_{\texttt{MDCP}}). \tag{1}$$

Now suppose that there exists an α-approximation algorithm A for MDCP, the output of this algorithm for the instance $I_{\texttt{MDCP}}$ is a set $\mathcal{P}$ of disjoint paths $P_j = (s, v_{S_j}, t)$, $j = 1, 2, \ldots, |\mathcal{P}|$, whose cardinality satisfies $|\mathcal{P}| = val_A(I_{\texttt{MDCP}}) \geq \frac{1}{\alpha} OPT(I_{\texttt{MDCP}})$. Consider the algorithm A' applied to $I_{\texttt{MSP}}$ which gives as output the family $\mathcal{S}' = \{S_1, S_2, \ldots, S_{|\mathcal{P}|}\}$ associated with $\mathcal{P}$. The family $\mathcal{S}'$ is a feasible solution for $I_{\texttt{MSP}}$, whose value is $val_{A'}(I_{\texttt{MSP}}) = |\mathcal{P}| \geq \frac{1}{\alpha} OPT(I_{\texttt{MDCP}})$ and by inequality (1) $val_{A'}(I_{\texttt{MSP}}) \geq \frac{1}{\alpha} OPT(I_{\texttt{MSP}})$ and so A' is an α-approximation algorithm for MSP.

Finally the first statement of the theorem follows from the $O(|S|^{1-\varepsilon})$ inapproximability of MSP, for any $0 < \varepsilon < 1$, and from the fact that $|V| = |\mathcal{S}| + 2$. Note that in the first coloring each color is associated with at most two edges which implies that EPC ≤ 2. The second statement follows from the fact that MSP is APX-hard if the cardinality of all sets in $\mathcal{S}$ is upper bounded by a constant $c \geq 3$ and that using the second coloring CPE $\leq c$. The results for DAGs come from the fact that the graph $K_{2,|\mathcal{S}|}$ obtained in the transformation can be oriented into a DAG in a straightforward way. □

Parameterized complexity

The next results are expressed in terms of *parameterized complexity* [12]. Recall that a problem with input size n is *fixed parameter tractable* with respect to some parameter k (and so is *in FPT*) if it can be solved in time $O(f(k) \cdot n^{O(1)})$ where the function f depends only on k. A problem is $W[1]$-hard if a $W[1]$-complete problem (e.g., deciding if the graph contains a clique of size k) can be reduced to it in FPT-time.

Theorem 6. MDCP *is $W[1]$-hard when parameterized by the number k of color-disjoint paths, even in DAGs and when the star property holds.*

Proof. It is enough to observe that the reduction used in Theorem 5 is a parameterized-preserving reduction where the parameter is the number of color-disjoint paths which corresponds to the number of disjoint subsets in a set packing. In [11] (see also [25, Chapter 11.4.2, pages 193–195]) it has been shown that MSP is $W[1]$-hard if the parameter is the number of disjoint subsets. □

The above theorem implies that, unless $FPT = W[1]$, MDCP is not in FPT when parameterized by the number of color-disjoint paths, that is there is no algorithm which finds k color-disjoint paths in $O(f(k) \cdot poly(|V| + |E|))$ time in DAGs, unless $FPT = W[1]$. Moreover, Theorem 1 shows that even finding a fixed number $k \geq 2$ of color-disjoint paths is NP-complete in general undirected graphs. This implies that MDCP is ParaNP-hard in undirected graphs, that is, it is impossible to devise an algorithm which finds k color-disjoint paths in $O((|V| + |E|)^{f(k)})$ time, unless $P = NP$.

The algorithm in Section 5.1 can be used to find an exact polynomial-time algorithm for MDCP when $|\mathcal{C}| = O(1)$. In fact, it is enough to search for the maximum k for which such algorithm returns k color-disjoint paths. As the maximum number of color-disjoint paths is upper bounded by Δ, we can use a binary search approach to solve the problem, applying at most $\log \Delta$ times the algorithm of Section 5.1. The next corollary follows.

Corollary 1. *The* MDCP *problem is FPT when parameterized by the number of colors* $|\mathcal{C}|$. *Moreover, there exists an algorithm for solving the* MDCP *problem in time* $O(f(|\mathcal{C}|)(|V|+|E|)\log\Delta)$, *where* f *is a function depending solely on* $|\mathcal{C}|$, *and* Δ *is the maximum degree of the graph.*

We can use the algorithms for bounded degree presented in Section 5.2 for solving MDCP when $\Delta \leq 4$. We get a polynomial-time exact algorithm for $\Delta \leq 3$ and a 2-approximation algorithm for $\Delta = 4$ as the maximum number of color-disjoint paths is upper bounded by $\Delta = 4$. Both algorithms require $O(|V|+|E|)$ time.

	Δ	EPC	CPE	k-DCP	MDCP
Undirected graphs	unbounded	1	unbounded	Solvable in $O(\lvert V\rvert+\lvert E\rvert)$	Solvable in $O(\Delta\lvert E\rvert)$
		unbounded	1	Solvable in $O(\lvert V\rvert+\lvert E\rvert)$ [7]	Solvable in $O(\Delta_{\mathcal{C}}\lvert E\rvert)$ [7]
	≥ 8	≥ 2	≥ 4	NP-hard for $\Delta \geq \max\{8,k\}$	Not approximable within $O(\lvert V\rvert^{1-\varepsilon})$, for any $0<\varepsilon<1$
		≥ 4	≥ 2		
	≤ 3	unbounded	unbounded	Solvable in $O(\lvert V\rvert+\lvert E\rvert)$	Optimum in $O(\lvert V\rvert+\lvert E\rvert)$
	$=4$	unbounded	unbounded	Solvable in $O(\lvert V\rvert+\lvert E\rvert)$ for $k=2$	2-approximation in $O(\lvert V\rvert+\lvert E\rvert)$
$\lvert\mathcal{C}\rvert = O(1)$, even without star	unbounded	unbounded	unbounded	Solvable in $O(f(\lvert\mathcal{C}\rvert)(\lvert V\rvert+\lvert E\rvert))$, in FPT when parameterized by $\lvert\mathcal{C}\rvert$	Optimum in $O(f(\lvert\mathcal{C}\rvert)(\lvert V\rvert+\lvert E\rvert)\log\Delta)$, in FPT when parameterized by $\lvert\mathcal{C}\rvert$
DAG	unbounded	≥ 3	≥ 3	Solvable in $O(\text{CPE}^2\lvert V\rvert\lvert E\rvert^{2k})$	NP-hard
		≥ 2	≥ 6		
		≥ 2	unbounded		Not approximable within $O(\lvert V\rvert^{1-\varepsilon})$, for any $0<\varepsilon<1$
		unbounded	$=3$		APX-hard
		unbounded	unbounded		$W[1]$-hard when parameterized by the number of paths

Table 2: Summary of complexity results.

7 Conclusion

Our results, presented in this paper and summarized in Table 2, give an almost complete characterization of the problem of finding SRLG-disjoint paths in networks with SRLGs satisfying the star property. For the case EPC $= 1$, the problem of finding k color-disjoint paths is equivalent to finding k edge-disjoint paths, thus a flow algorithm such as the Ford-Fulkerson's [15] can be used to solve k-DCP in time $O(|V|+|E|)$ and MDCP in time $O(\Delta|E|)$. As for the case CPE $= 1$, every edge has one color and since the star property is satisfied, all colors have span 1 (i.e. an edge has only one color and the set of edges having the same color forms a connected component). The problem of finding color disjoint-paths in graphs with span 1 has been proven polynomial in [7]. To conclude, we point out some open questions for further research:

- The complexity of the problem is still open for the cases where the maximum degree of the network is equal to 5, 6 or 7 and for the cases where EPC $\in \{2,3\}$ and CPE $\in \{2,3\}$. Solving these cases will give a complete complexity characterization of the problem with respect to the maximum degree of the network and the parameters EPC and CPE.
- In the problem definition, we assumed that $\mathscr{C}(e) \neq \emptyset$, for each $e \in E$, which means that each edge of the network must have at least one color associated with. If we allow edges with no colors, then color-disjoint paths are not necessarily edge-disjoint, since an edge

that is not affected by any SRLG can be shared by any number of paths. Therefore, an interesting direction would be to constrain the paths to be edge-disjoint as well; so, the problem would be to find k SRLG-disjoint edge-disjoint paths in the case where the number of SRLGs is bounded by a constant and the number of edges with no SRLG is linear in the size of the network. This does not affect the hardness results that hold for the (restricted) case where each edge has a color and hence also for the (more general) case where colorless edges are allowed. However, the polynomial-time algorithms proposed in Section 5 do not work in this case.

- It would be interesting to consider the unsolved cases for directed graphs since we only have results for the specific case of DAGs so far.

Acknowledgments

We thank the three referees for their very helpful comments and remarks which improved the quality of the article.

References

[1] G. Ausiello, A. D'Atri, and M. Protasi. Structure preserving reductions among convex optimization problems. *J. Comput. Syst. Sci.*, 21(1):136–153, Aug. 1980.

[2] R. Bhandari. *Survivable Networks: Algorithms for Diverse Routing.* Kluwer Axademic Publishers, 1998.

[3] A. Björklund, T. Husfeldt, and M. Koivisto. Set partitioning via inclusion-exclusion. *SIAM J. Comput.*, 39(2):546–563, 2009.

[4] H. Broersma, X. Li, G. Woeginger, and S. Zhang. Paths and cycles in colored graphs. *Australas. J. Combin.*, 31:299–311, 2005.

[5] R. D. Carr, S. Doddi, G. Konjevod, and M. Marathe. On the red-blue set cover problem. In *Proceedings of the Eleventh Annual ACM-SIAM Symposium on Discrete Algorithms*, SODA '00, pages 345–353, Philadelphia, PA, USA, 2000. Society for Industrial and Applied Mathematics.

[6] V. T. Chakaravarthy. New results on the computability and complexity of points–to analysis. In *ACM SIGPLAN-SIGACT symposium on principles of programming languages (POPL)*, pages 115–125. ACM, 2003.

[7] D. Coudert, P. Datta, S. Perennes, H. Rivano, and M.-E. Voge. Shared risk resource group: Complexity and approximability issues. *Parallel Processing Letters*, 17(2):169–184, June 2007.

[8] D. Coudert, S. Pérennes, H. Rivano, and M.-E. Voge. Combinatorial optimization in networks with Shared Risk Link Groups. Research report RR-8575, Inria, July 2014.

[9] P. Datta and A. Somani. Graph transformation approaches for diverse routing in shared risk resource group (SRRG) failures. *Computer Networks*, 52(12):2381–2394, Aug. 2008.

[10] J. Doucette and W. Grover. Shared-risk logical span groups in span-restorable optical networks: Analysis and capacity planning model. *Photonic Network Communications*, 9(1):35–53, 2005.

[11] R. G. Downey and M. R. Fellows. Fixed-parameter tractability and completeness II: On completeness for W[1]. *Theoretical Computer Science*, 141(1-2):109–131, 1995.

[12] R. G. Downey and M. R. Fellows. *Fundamentals of Parameterized Complexity.* Texts in Computer Science. Springer, 2013.

[13] A. Faragó. A graph theoretic model for complex network failure scenarios. In *8th INFORMS Telecommunications Conference*, Dallas, Texas, Mar. 2006.

[14] M. R. Fellows, J. Guob, and I. Kanj. The parameterized complexity of some minimum label problems. *Journal of Computer and System Sciences*, 76(8):727–740, Dec. 2010.

[15] L. R. Ford and D. R. Fulkerson. A simple algorithm for finding maximal network flows and an application to the hitchcock problem. *Canadian Journal of Mathematics*, 9:210–218, 1957.

[16] M. Garey and D. Johnson. *Computers and Intractability: A Guide to the theory of NP-completeness.* Freeman NY, 1979.

[17] L. Guo and L. Li. A novel survivable routing algorithm with partial shared-risk link groups (SRLG)-disjoint protection based on differentiated reliability constraints in wdm optical mesh networks. *IEEE/OSA Journal of Lightwave Technology*, 25(6):1410–1415, June 2007.

[18] R. Hassin, J. Monnot, and D. Segev. Approximation algorithms and hardness results for labeled connectivity problems. *Journal of Combinatorial Optimization*, 14(4):437–453, Nov. 2007.

[19] J. Hopcroft and R. Tarjan. Algorithm 447: Efficient algorithms for graph manipulation. *Commun. ACM*, 16(6):372–378, June 1973.

[20] J. Håstad. Clique is hard to approximate within $n^{1-\varepsilon}$. *Acta Mathematica*, 182(1):105–142, 1999.

[21] J. Hu. Diverse routing in mesh optical networks. *IEEE Transactions on Communications*, 51:489–494, Mar. 2003.

[22] Q. Jiang, D. Reeves, and P. Ning. Improving robustness of PGP keyrings by conflict detection. In *RSA Conference Cryptographers' Track (CT-RSA)*, volume 2964 of *LNCS*, pages 194–207. Springer, Feb. 2004.

[23] V. Kann. Maximum bounded 3-dimensional matching is MAX SNP-complete. *Inf. Process. Lett.*, 37(1):27–35, Jan. 1991.

[24] X. Luo and B. Wang. Diverse routing in WDM optical networks with shared risk link group (SRLG) failures. In *Proc. DRCN*, pages 445–452. IEEE, Oct. 2005.

[25] R. Niedermeier. *Invitation to fixed-parameter algorithms.* Oxford lecture series in mathematics and its applications. Oxford University Press, 2006.

[26] L. Shen, X. Yang, and B. Ramamurthy. Shared risk link group (SRLG)-diverse path provisioning under hybrid service level agreements in wavelength-routed optical mesh networks. *IEEE/ACM Transactions on Networking*, 13:918–931, Aug. 2005.

[27] J. W. Suurballe. Disjoint paths in a network. *Networks*, 4(2):125–145, 1974.

[28] J. W. Suurballe and R. E. Tarjan. A quick method for finding shortest pairs of disjoint paths. *Networks*, 14(2):325–336, 1984.

[29] S. Szeider. Finding paths in graphs avoiding forbidden transitions. *Discrete Applied Mathematics*, 126(2–3):261–273, 2003.

[30] A. Todimala and B. Ramamurthy. IMSH: an iterative heuristic for SRLG diverse routing in WDM mesh networks. In *IEEE International Conference on Computer Communications and Networks (ICCCN)*, pages 199 – 204. IEEE, 2004.

[31] A. Todimala and B. Ramamurthy. Survivable virtual topology routing under shared risk link groups in WDM networks. In *International Conference on Broadband Communications, Networks and Systems (BroadNets)*, pages 130–139, San Jose, CA, USA, Oct. 2004. IEEE.

[32] S. Yuan, S. Varma, and J. Jue. Minimum-color path problems for reliability in mesh networks. In *Proc. IEEE INFOCOM*, volume 4, pages 2658–2669, Houston, TX, USA, Mar. 2005.

[33] Q. Zhang, J. Sun, G. Xiao, and E. Tsang. Evolutionary algorithms refining a heuristic: A hybrid method for shared-path protections in WDM networks under SRLG constraints. *IEEE Transactions on Systems, Man and Cybernetics, Part B*, 37(1):51–61, Feb. 2007.